



CSC REPORT



THE DESIGN OF DATA PROCESSING COMPILERS FOR LARGE SCALE COMPUTERS



THE DESIGN OF DATA PROCESSING COMPILERS FOR LARGE SCALE COMPUTERS



CSC Vice President Roy Nutt and Senior Scientist Dr. Charles J. Swift

By ROY NUTT and DR. CHARLES J. SWIFT

Design of a data processing compiler is first begun by examination of the language which the compiler will be constructed to translate. It is unfortunate, but true, that the so-called machine independent common languages of today are incompletely defined, and it falls to the implementor of a compiler for a specific machine to discover or rediscover such incompletenesses and substitute definitions acceptable to his customer and the eventual users of the compiler.

Among the principal considerations are the definition of arithmetic rules, the collation sequence, the characteristics of auxiliary storage with respect to random access or directional behavior, limitations on identifier length in construction of the source program and, of course, the character set which will be accepted by the compiler or its object programs. Choices must be made which maintain a reasonable degree of compatibility with other compilers without unduly compromising efficiency.

Consider the apparently simple choice of identifier length for a COBOL compiler. The language definition presented by DOD documentation suggests, but does not require, recognition of a 30-character identifier. This choice is awkward for most word-organized computers of today (24 characters would be more convenient as it is divisible by 4, 6 and 8), and wasteful if fixed length representation is used. However, the language definition permits use of an arbitrarily long identifier and to maintain compatibility with all possible implementations of COBOL, such identifiers must be retained in their entirety. Complexities attendant to manipulation of variable length data are acceptable if only internal processing is required although when extended to auxiliary storage, they are magnified to an almost unreasonable degree. Furthermore, as few compilers have been constructed for less than ten dollars per instruction, the inclusion of completely general processing of COBOL identifiers may increase cost of the compiler by many thousands of dollars.

Another major problem associated with the language specifications arises from over-definitions so that when explicitly followed, the result may seriously impair the efficiency of object programs and increase the cost of the compiler; if not followed, an incompatibility is introduced. An example is the COBOL requirement that minus zero be everywhere replaceable and equivalent to plus zero, thus causing comparison tests to be much more time consuming than would otherwise be the case. Since object programs can usually be constructed to avoid generating minus zero internally, the designer must decide whether the cost to the user is worth the small measure of compatibility obtained by adhering to this specification.

Attention must also be given to the environment in which the compiler and its object programs will function. Most large scale computers of today utilize an executive monitor or operating system to control the execution of individual programs, and such systems usually restrict the manner in which various components may be used—in some cases, they even prohibit the use of certain units or functions of the basic hardware. The design of the compiler must take cognizance of monitor restrictions which affect compilation, and restrictions (sometimes different) which affect object programs. For example, a data processing program often requires extensive communication with the computer operator concerning such functions as the mounting and dismounting of tapes, establishment of rerun points, etc. Such communications often are in the domain of the monitor, and if all requirements are not satisfied, independent or intermediate routines must be designed to overcome deficiencies.

Availability of a monitor may be a boon to the compiler implementor, since functions such as general input/output subroutines must otherwise be supplied by him. However, the monitor often introduces complications. In the unlikely event that the monitor is already functioning, available docu-

mentation may not adequately cover the use and operation in the detail required; or as is more usual, it is not yet operational and the construction of the compiler is hampered by both the lack of sufficient documentation and the concurrent debugging of the monitor. Those programmers who have participated in a joint effort of any size are quite familiar with the problem of isolating the source of error when two incomplete systems fail to operate correctly together.

As the discovery and correction of errors is of particular importance, the design of the compiler must provide for maximum assistance to the user in these areas. Thorough diagnosis of syntactical and semantical errors should be made, and reported in a manner which will facilitate correction (e.g., by indicating source line number). Careful consideration of the compiling process may indicate other ways in which useful debugging information can be supplied to the user.

Memory limitations imposed by the smaller configurations of the computer usually dictate selection of a multi-pass compiler to satisfy the need for efficiency in a data processing program. Whereas the ordinary scientific application may often be accomplished by a one-shot program, it is far more usual for the data processing application to require a program which will be repetitively executed on a daily or perhaps an hourly basis. At this stage in the development of compilation techniques, methods of optimization in a limited memory by a one-pass system are insufficient. A multi-pass system, however, affords opportunities for examination and re-examination of the program which permit optimizations of several kinds. Chief among these is the classification of similar object subprogram steps, such as those required for a "move", whereby the compiler can construct tailored subroutines for the more frequently encountered cases. Still another consideration is the usually large number of instructions and wide variety of input and output options

offered by a large scale computer, which should be utilized wherever their flexibility would enhance the object program. Despite these considerations, in the event that there are limitations on expenditures or time scale which preclude the development of a sophisticated system, a one-pass compiler may be indicated.

Finally, the design and construction of the compiler must be modular to permit incorporations due to language changes, computer changes, or corrections. Above all, it is necessary to carefully document all aspects of its structure. Early documentation will greatly increase the likelihood of timely completion, reduce the total cost of construction and, if regularly updated, assure final documentation at an early date. Furthermore, proper documentation will avoid the catastrophic delays sometimes caused by staff changes which are not anticipated.

It should be noted that comprehensive data processing languages and data processing compilers are relatively new and many changes can be expected in the future. New features are being added to COBOL; FORTRAN has been modified to enhance its data processing capabilities, and the application of other basic types of languages (such as decision tables) to data processing will continue to influence the field. As users of data processing equipment increase their sophistication, we may expect a change in languages away from traditional terminology to more formal logical terminology. Present research into the nature of syntactic analysis also lends promise of generalized compilers that can process any of a large class of languages, thereby enabling the language to be easily changed to suit a particular job in hand.

In the near future, perhaps within five years, developments such as these will ease the design and construction of data processing compilers by providing the systems programmer with tools similar to those he has so long constructed for others.

JPL AWARDS CSC NEW SFOF CONTRACT

The California Institute of Technology's Jet Propulsion Laboratory last month awarded CSC a major contract in excess of \$400,000.

The contract provides for a continuing CSC effort on JPL's new Space Flight Operations Facility (SFOF) which will be used to control NASA unmanned lunar and interplanetary space probes. Modifications to the SFOF system to be accomplished by CSC during the coming year will support Mariner and Surveyor spacecraft launchings in 1964 and 1965.

CSC has performed services for Jet Propulsion Laboratory since May, 1961.

CSC WINS SEVEN CONTRACTS FROM LOCKHEED MISSILES & SPACE

Seven new contracts for programming services and support were received by CSC last month from the Lockheed Missiles & Space Company, Sunnyvale, Calif.

The Lockheed awards to CSC are projected to be in excess of \$1,000,000. CSC will provide computer programming support for such large scale equipment as the IBM 7090/7094 as part of an Automatic Data Accumulation (ADA) system. The seven new awards from Lockheed are in addition to a major contract from Lockheed announced in mid-September.

The ADA complex is part of an overall management and information control system and is considered a major cost cutting weapon. It is essentially a communications and retrieval system which enables production workers at remote locations to continuously add data bit by bit as it is created and to retrieve it on demand. Most of the data processing is accomplished in batches with information collected until there is a sufficient amount to permit efficient computer operation.

Programming by CSC will be accomplished in COBOL by personnel from CSC's new San Francisco Bay Area office located in Palo Alto.

RECORD SALES AND EARNINGS PROJECTED FOR FY 64

A record year for sales volume and net earnings was projected last month by CSC President Fletcher Jones.

Mr. Jones explains, "CSC is now developing proprietary programs for repetitive sale to business and scientific users of computer equipment. This, combined with the recent acquisition of several large contracts, is expected to provide a substantial effect on service revenues and net earnings."

In other areas of projected expansion, Mr. Jones adds that CSC is currently considering several acquisition possibilities as part of its growth program. Mr. Jones states that companies with whom negotiations for acquisition are presently underway

are highly profitable and in the same general area of activity as CSC.

For the mid-year ending September 30, a record high in sales and earnings was recently announced by CSC.

Service revenues during the six month period were \$1,968,657, an increase of 49 percent over \$1,318,115 for the corresponding period in 1962. Net earnings also increased from \$144,923 to \$201,254, and per share earnings rose from 23 cents to 31 cents based on 640,000 shares outstanding.

COMMERCIAL APPLICATIONS RECEIVES LITTON CONTRACT

CSC's Commercial Applications Department was awarded a contract for programming services last month by the Guidance and Control Systems Division of Litton Industries.

The Litton contract provides for the design and reprogramming of a major cost accounting system. Programming will be performed in COBOL for various IBM computers.

MAJOR RECRUITING DRIVE INITIATED BY CSC

As a result of several new contracts recently received, CSC has undertaken a nationwide recruiting program expected to increase by 50 percent the company's present staff of approximately 200 analysts and programmers.

In national trade publications and major cities throughout the United States, CSC has been advertising for personnel experienced in commercial applications, scientific and systems programming. Openings presently exist in all offices of the company for programmers with a degree and three or four years experience.

EIGHT MANAGERIAL APPOINTMENTS ANNOUNCED AT CSC

The establishment of two new Departments and six new Groups as well as the appointment of their respective managers was announced last month at CSC.

A newly created Department of Production Scheduling, Standards and Quality Control will be managed by Robert Paul, while Joel Erdwinn has been appointed Manager of the Programming Research and Development Department.

Mr. Paul joined CSC in June, 1959, and has been a project group manager supervising several major systems programming efforts. He was formerly with the Rocketdyne and Columbus divisions of North American Aviation.

Mr. Erdwinn joined CSC in July, 1960, and has been responsible for the design and implementation of a variety of programming systems. He was formerly a mathematician at Shell Oil Co. and a programming system designer for Burroughs Corp.

In Scientific Applications, Steve Hornick has been appointed Real Time Systems Group Manager; Valmer Norrod, Engineering Applications Group Manager; Paul Kazek, Special Projects Group Manager; and Dr. Allan Harbaugh, Operations Research Group Manager.

Mr. Hornick is a veteran of seven years in the real time field and has been project leader of the CSC team for development of a command and control system in support of Jet Propulsion Laboratory's Space Flight Operations Facility. He was formerly

with Systems Development Corporation and Douglas Aircraft.

Mr. Norrod has directed programs for video data reduction and analysis and was formerly with Ling Temco Vought as scientific computing supervisor.

Mr. Kazek has extensive experience in analysis and design of computer solutions to nuclear weapon system problems and was formerly with Los Alamos Scientific Laboratory.

Dr. Harbaugh is well known in the field of operations research and has more than 15 years professional experience. He was formerly with Space Technology Laboratories, Space General Corporation, The Rand Corporation, and Defense Research Laboratory.

In Systems Programming, Dale Hanks has been appointed Utility Systems Group Manager and Lou Gatt, Scientific Systems Group Manager.

Mr. Hanks has more than ten years experience in systems programming and engineering applications. He joined CSC in August, 1959, after seven years with North American Aviation.

Mr. Gatt supervised the development of FORTRAN IV for the UNIVAC 1107 as well as several other major computer systems. He was formerly with Los Alamos Scientific Laboratory and Douglas Aircraft.

Appointed to the position of Senior Scientist at CSC are Dr. H. Gordon Rice and Dr. Charles Swift.

Dr. Rice joined CSC in October, 1960, and has more than 12 years professional experience in numerical analysis and scientific applications. He has accomplished original research in recursive function theory and artificial languages.

Dr. Swift joined CSC in August, 1959, and has more than 12 years experience in numerical analysis and computer applications. He was instrumental in the design of the FACT language and compiler for the H-800 and COBOL for the UNIVAC 1107.

NEW INPUT TO CSC ADDS 74 MAN YEARS

Ten new technical staff members joined CSC last month adding more than 74 man years experience to the company in Commercial and Scientific Applications.

Russell Unfreed, Jr., formerly of Thompson Ramo Wooldridge, and William Frost of Aerospace Corp. were appointed to the Scientific and Military Applications Department in Los Angeles. Donald M. Hughes of Hughes Aircraft has joined the Commercial Applications Department.

Assigned to CSC's Palo Alto office are Lewis H. Busell from Control Data Corporation; Robert M. Eve, formerly a consultant in business systems analysis; Thomas R. Fosnight from North American Aviation; Ralph M. Jones from Lawrence Radiation Laboratory; Robert S. Long from the Boeing Company; Richard D. Miller, Astronautics Division of General Dynamics; Ronald F. Rykowski from Data Reduction Consultants, and David A. Stevenson from Philco Corp.

CSC REPORT — DECEMBER, 1963
Published by Computer Sciences Corporation
650 North Sepulveda Boulevard
El Segundo, California
Los Angeles/San Francisco/Houston/New York/London